

SantaFe_Edison Web Development Document

프로젝트명	SantaFe
담당교수	서영균
작성자	오정석



차 례

01 문서 소개

02 사용 기술 스택

03 Front-end

04 Back-end



■ 문서 소개

최근 하드웨어의 성능 향상과 데이터 수집 기술의 발달, 공공데이터의 개방 등의 요인으로 계산과학 분야에서 대용량의 데이터가 축적되었으며, 이를 바탕으로 매우 다양한 데이터 분석활동이 진행되고 있다. 하지만 연구자들의 분석활동으로 만들어진 결과물들은 대부분이 문석, 소스 파일의 형태로 존재한다. 이렇듯 결과물은 프로그램의 형태를 띄고 있지 못하므로 결과를 사용하고자 하는 사용자들에게 많은 제약과 불편함이 따른다.

본 SantaFe 프로젝트는 이러한 제약, 불편을 해소하기 위해 수집 데이터를 기반으로 사용자에게 시뮬레이션 작업 환경을 제공하는 프레임워크를 개발하고자 하는 취지를 가진다. 본 문서는 이러한 SantaFe 프레임 워크의 유저 인터페이스 측면인 웹 시스템의 개발 내용을 명시하기 위해 작성되었다.

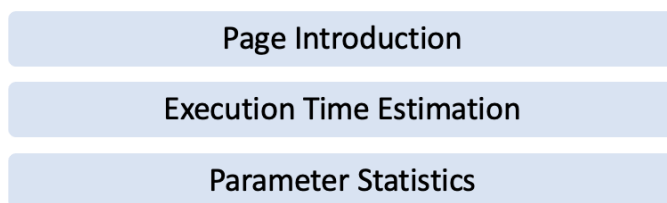
■ 사용 기술 스택

Development Part	Tech Stack
Client	React
Server	Node.js, Express.js

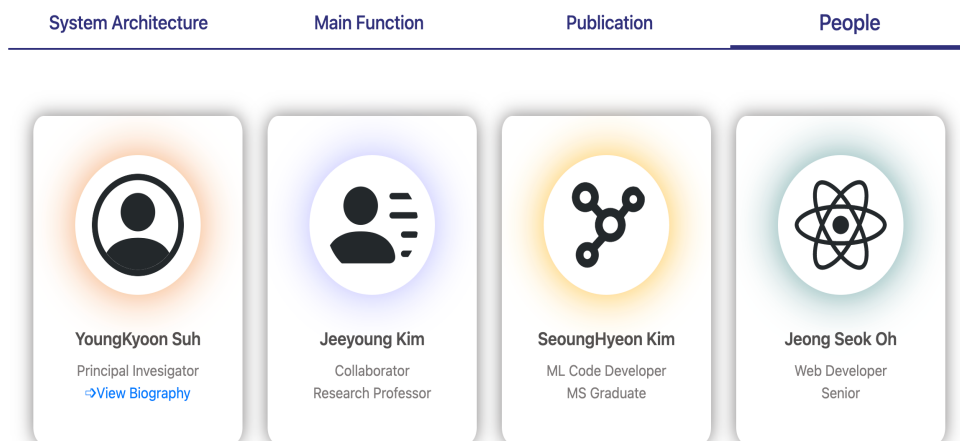
본 웹 시스템은 Client단의 경우 React 라이브러리를 중심으로 개발이 진행되었다. Client 코드는 javascript에 대한 이해가 필요하며 React Component, Router, Life Cycle에 관한 전반적인 이해를 필요로 한다. Server단의 경우 Node.js와 Express 패키지에 관한 기본적인 이해를 필수로 한다. 또 Server와 Client간에 Rest API를 이용해 통신이 이루어지기 때문에 Rest API에 관한 이해가 필요하다.

■ Front-end

페이지 구성



본 웹 시스템은 크게 3가지의 페이지로 구성된다. 첫 번째는 Page Introduction 페이지로 시스템 구성 소개, 주요 기능, Publication, 개발 인원 소개 내용을 포함하고 있다. 두 번째는 Execution Time Estimation 페이지로 App을 선택하고 parameter를 입력 시 시뮬레이션 시간을 추정해 볼 수 있는 기능을 가진다. 세 번째는 Parameter Statistics 페이지로 이용자들의 검색 데이터들을 바탕으로 자주 검색해 본 Parameter set 결과를 확인해 볼 수 있다.



Page Introduction 페이지의 경우 horizontal Menu bar로 세부 콘텐츠들을 확인해볼 수 있게끔 구성되어 있다.

Simulation Execution-Time Estimation Service

Choose a computational science and engineering area

Computational Thermal Fluid

Nanophysics

Calculation Chemistry

Computational Medicine

NEXT

Execution Time Estimation 페이지는 Computational Science and engineering area 선택, App type 선택, Parameter 입력 세 번의 세부 폼을 거친 후 simulation time을 추정해 볼 수 있다. Time estimation 페이지의 경우 세 번의 선택 페이지가 라우터를 통해 구분된 것이 아닌 세부 폼 영역만 컴포넌트로 분할해 교체해주고 있는 방식을 사용했기 때문에 뒤로 가기 버튼으로 폼의 이동은 불가능 하다. 대신 하단의 Back, Next, Submit 버튼을 통해 세부 폼을 이동, 제출할 수 있다.

세부 선택 페이지는 아무런 항목을 선택하지 않았거나 파라미터 항목 중 하나라도 입력되지 않았을 경우에는 다음 페이지로 이동할 수 없도록 처리가 되어 있다.

Simulation Execution-Time Estimation Service

Science & Engineering Type
Nanophysics

Appname
gravityslshot

Parameters
in01 : 1 in02 : 0.5 in03 : 3
in04 : 1

Estimation Time
585.2815 sec

Estimation 결과는 좌측 선택 및 입력 결과와 우측 시뮬레이션 추정 시간을 다음과 같이 출력한다. 다시 Estimation 페이지로 돌아가기 위해선 하단의 RESET 버튼을 사용한다.

Parameter Ranking Statistics Service

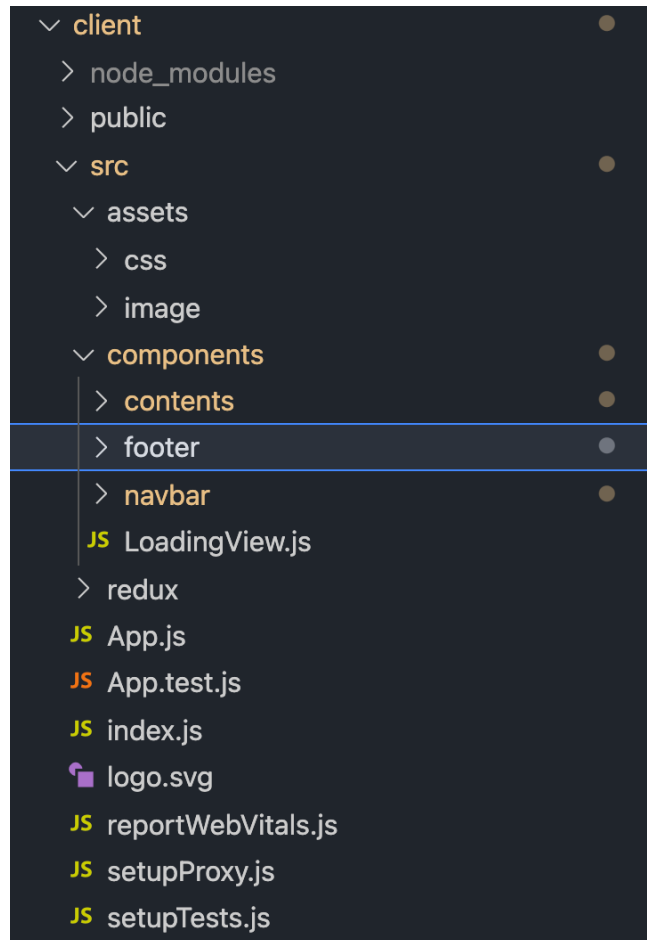
Cluster Type : Computational Medicine
App Type : Single_Cell_Electrophysiology

RANKING	in01	in02	in03	in04	in05	in06	in07	in08	in09	in10	in11	in12	in13	in14	in15
1	100000	0	tn2006	ori	1	0.05	30	1000	30	88000010	600	0.1	-73	-10	1.8
2	100000	0	tn2006	ori	3	0.05	30	1000	30	88000010	600	0.1	-73	-10	1.8
3	100000	0	tn2006	ori	2	0.05	30	1000	30	88000010	600	0.1	-73	-10	1.8
4	40000	0	tn2006	V241F	1	0.05	20	1000	30	88000010	600	0.1	-73	-10	1.8
5	100000	0	tn2006	ori	1	0.05	30	1000	30	100	600	0.1	-73	-10	1.8
6	100000	0	tn2006	ori	1	0.05	30	1000	20	100	600	0.1	-73	-10	1.8
7	100000	0	tn2006	V241F	1	0.05	30	1000	30	88000010	600	0.1	-73	-10	1.8
8	100000	0	tn2006	ori	3	0.05	30	1000	30	100	600	0.1	-73	-10	1.8
9	100000	0	tn2006	ori	2	0.05	30	1000	30	100	600	0.1	-73	-10	1.8
10	100000	0	tn2006	V241F	2	0.05	30	1000	30	88000010	600	0.1	-73	-10	1.8

[RESET](#)

Parameter Statistics 페이지의 경우, Computational Science and engineering area 선택, App type 선택, Parameter 입력 폼은 Execution Time Estimation 페이지와 동일하다. 검색하고 싶은 항목과 파라미터를 입력할 시 위 화면과 같이 가장 많이 검색된 상위 10개의 parameter set이 화면에 출력된다.

Component 구조



React 디렉토리 구조는 다음과 같다. assets 디렉토리에서는 css 파일과 사용 image 파일들이 포함되어 있다. component 디렉토리는 본 웹 페이지를 구성하는 컴포넌트 파일들이 포함되어 있다. component는 크게 navbar, footer, contents로 분할되어 있다. LoadingView의 경우 페이지 로딩 시 보여지는 애니메이션 페이지이다. content 컴포넌트는 home, introduction, estimation, statistics와 같이 페이지 별 컴포넌트들로 구분되어 있다. 아래쪽의 setupProxy.js는 현재 React와 Node.js를 연동하는데 프록시 서버를 활용하고 있는데 그에 해당하는 코드이다.

■ Back-end

서버 구성

서버 구성은 node.js와 express를 활용하였으며 데이터베이스는 mongoDB를 사용하였다. 그리고 Simulation Estimation의 경우 별도로 cmd를 호출하여 Rscript를 실행하는 방식이기 때문에 Rscript를 이해하기 위해선 R의 전반적인 이해도가 필요하다. 이 외에도 cmd 실행, DB 쿼리 실행 등에 Promise, async await 형태의 코드가 사용되기 때문에 비동기 코드 구조를 이해할 필요가 있다.

API 명세

REST API Table
Science and Engineering Area List API
Science App Name List API
Parameter List API
Estimation Result API
Parameter Statistics API

API는 크게 4가지로 구성된다. Science and Engineering Area 타입들을 반환하는 Science and Engineering Area List API와 app name 종류들을 반환하는 Science App Name List API, 해당 App type의 파라미터 종류들을 반환하는 Parameter List API, 시뮬레이션 추정 시간을 반환하는 Estimation Result API, 해당 요청 Parameter 데이터 셋을 반환하는 Parameter Statistics API로 구성된다.

1. Science and Engineering Area List API

- Method : GET
- URI : /api/clusters
- 설명 : Cluster(Science and Engineering Area) 타입 제공
- Response

```
[
  "Computational Thermal Fluid",
  "Nanophysics",
  "Calculation Chemistry",
  "Computational Medicine"
]
```

2. Science App Name List API

- Method : GET
- URI : /api/appnames/{cluster}
- 설명 : Science App name 종류 리스트 제공, {cluster}는 직전 페이지에서 선택한 cluster 타입이다.
- Response (ex : /api/appnames/Nanophysics)

```
[
  "acuteSTMtip",
  "BAND_DOSLab",
  "coulombdart",
  "gravityslingshot",
  "LCAODFTLab",
  "PhaseDiagramSW",
  "pianostring",
  "roundSTMtip",
  "WaveSimulation"
]
```

3. Parameter List API

- Method : GET
- URI : /params/{cluster}/{appname}
- 설명 : 선택한 App 타입에 해당하는 파라미터 종류 리스트를 제공. {cluster}는 선택한 cluster 타입을, {appname}은 선택한 Appname 타입을 가리킴.
- Response (ex : /api/params/Nanophysics/coulombdart)

```

1  [
2      "in01",
3      "in02",
4      "in03",
5      "in04",
6      "in05",
7      "in06",
8      "in07",
9      "in08",
10     "in09",
11     "in10",
12     "in11",
13     "in12"
14 ]

```

4. Estimation Result API

- Method : POST
- URI : /api/estimate-result
- 설명 : 선택한 cluster, appname, parameter을 시뮬레이션한 실행 시간을 제공
- Response : execution time을 string 형태로 제공

5. Parameter Statistics API

- Method : GET
- URI : /api/statistics/{appname}
- 설명 : 선택한 appname의 상위 검색 parameter set 10개 제공
- Response : 아래와 같은 하나의 데이터 셋 형태 10개 반환

```

{
  "params": [
    "-0.5",
    "5"
  ],
  "_id": "60471541bf3b69068bca6b66",
  "cluster": "Nanophysics",
  "appname": "acuteSTMtip",
  "count": 363,
  "__v": 0
},

```


데이터베이스

Collection Name ^	Documents	Avg. Document Size	Total Document Size	Num. Indexes	Total Index Size	Properties
estimationtypes	19	247.5 B	4.6 KB	1	36.0 KB	
statisticdatas	51,367	214.3 B	10.5 MB	1	516.0 KB	

데이터베이스는 두 개의 Collection을 가진다. estimationtypes 컬렉션의 경우 웹 사이트에서 시뮬레이션할 수 있는 App의 종류에 대한 정보를 담고 있다. statisticdata 컬렉션의 경우 사용자들이 검색해본 parameter set들의 대한 정보를 담고 있다.

```
_id: ObjectId("60462c1fdeb7fe055aa470f5")
cluster: "Computational Thermal Fluid"
appname: "2D_Comp_P"
> params: Array
```

estimationtypes의 도큐먼트 한 예이다. 도큐먼트에는 cluster, apptype과 해당하는 parameter 종류들을 Array 형태로 저장하고 있다.

```
_id: ObjectId("6047114a0c219a05a253d117")
v params: Array
  0: "1"
  1: "1"
  2: "0.5"
  3: "11965932"
  4: "4"
  5: "0"
  6: "30000"
  7: "0.1"
  8: "10"
  9: "0.25"
  10: "0"
  11: "0.00001"
  12: "3"
  13: "6"
  14: "6"
  15: "3"
  16: "1"
cluster: "Computational Thermal Fluid"
appname: "2D_Comp_P"
count: 1
__v: 0
```

statisticdatas의 도큐먼트 한 예이다. 사용자가 입력한 params Array와 선택한 cluster, app type을 담고 있으며, 해당 파라미터 셋이 지금까지 몇 번 검색되었는지 횟수를 세고 있는 count 정보를 담고 있다.